

## Research on the Sentiment Analysis of Customer Reviews Based on the Ontology of Phone

Can YANG<sup>1, a</sup>, Zhiye CHEN<sup>2, b</sup>, Tianyi WANG<sup>3, c</sup>, Pengfei SUN<sup>4, d</sup>

<sup>1</sup>Department of Software Engineering, Beijing University of Posts and Telecommunications, Yang Can, 102209, China

<sup>2</sup> Department of Software Engineering, Beijing University of Posts and Telecommunications, Chen Zhiye, 102209, China

<sup>3</sup> International School, Beijing University of Posts and Telecommunications, Tianyi Wang, 102209, China

<sup>4</sup> Department of Software Engineering, Beijing University of Posts and Telecommunications, Pengfei Sun, 102209, China

<sup>a</sup>email: chomyangcan@163.com, <sup>b</sup>email: adianadian@126.com

<sup>c</sup>email: 1wangtianyi@sina.com, <sup>d</sup>email: sunpengfei@bupt.edu.cn

**Keywords:** Product review, Ontology, Feature-sentiment mining, Polarity computation

**Abstract.** In this paper, we develop FOBPRM (Feature Ontology Based Product Review Miner) system, to semi-automatically build the ontology tree of Phone area and extract the most representative expressions and customer opinions in the reviews, which represents for feature-sentiment pairs. Finally we develop our method of polarity calculation of feature-sentiment pairs and generate the all-round summary for customers and vendors. Instead of putting the emphasis on feature extraction and sentiment classification as the existing work did, we focus on the association between the features and sub-features of a product and their associated sentiment that influence the polarity of the attributes in fine-grained in this paper. Ontology built by computation of special degree and similarity degree has improved the accuracy and recall rate of features, and the information entropy computes the polarity of feature-sentiment pairs. The whole system works out the desired result.

### Introduction

With the development of the electronic commerce, more and more people are getting used to buying something they need via the B2C (Business to Consumer) platform. At the same time, the review systems can provide valuable customer feedback to not only the manufactures who want to know the current performance evaluation of the product, but also the potential customers who want to refer to other user's experience before purchasing.

However, the number of review grows so fast that it can make it difficult for users and merchants to read through it. There must be a automatic approach to actualize this.

Most of the existing work on mining customer reviews focuses on opinion feature extraction and adjective orientation identification. For example, a review summarization system [1] can extract nouns in comments using association mining and determine the orientation of the nearby adjective words using the information of synonyms and antonyms in WordNet [2]. Their system finally lists all the positive sentences and negative sentences with respect to each product feature. Many recent efforts have been made to improve the accuracy of feature extraction and sentiment analysis [3]. However, there are still some limitations of these existing review summarization systems. First, the accuracy of current automatic feature extraction methods is low (around 0.5–0.7 in precision). It is thus impractical to be put in use in real applications. Second, identification of the opinion words orientation is not satisfactory. As a consequence, the results of statistics on positive and negative opinions are unreliable. Third, current systems usually present their results as a list of sentences or

terms. When there is a large number of such form of result, it will be difficult for the customers to distinguish the specific useful comments for themselves.

To solve the above problems, we propose to review the summarization system, to automatically extract the most representative expressions and customer opinions in the reviews on various product features. Different from many other systems which use benchmark datasets, our system is a real practical system integrating review crawling from Jindong.com, automatic product feature extraction along with a text field where users can input their desired features. And then the summary can be generated using the information from a filed feature-sentiment ontology tree which is constructed from reviews. Finally, the most representative pair of feature and sentiment and the corresponding professional parameters are selected in the form of summarization.

## **Related Work**

Having researched some papers in product customer reviews mining, we found that mining and summarizing reviews involve three tasks: feature identification, sentiment analysis, and summarization. To be specific, feature identification aims at important product features; sentiment analysis is usually involved to identify if an opinion feature or sentence is positive or negative; and summarization is supposed to show the representative results to users. Though traditional sentiment analysis or opinion mining was performed at the document level [4], increasingly more research has examined opinion mining at the more fine-grained sentence or phrase level in recent years [5]. Therefore, product user reviews opinion mining is more usual performed at the product feature level to provide deep analytics for the target product, which is more fine-grained. The fact is that research on Chinese Customer Review Mining is at shortage with more difficulty analyzing Chinese sentiment and opinion. At the same time, features presenting as nouns have the hierarchical organization and sentiment words are often context-dependent. For instance, while the term “small” in the expression “the phone’s power consumption is small” implies a positive sentiment, the same term in the expression “the phone’s screen is very small” may have negative sentiment. Moreover, mining based-features and the nearby context sentiment is a key step to improve the results precision and accuracy with more knowledge digging out. In this context, a domain ontology to structure and extract product features as well as to produce a comprehensive summary has been proposed in Chinese reviews.

Feature-based opinion mining is the research problem that focuses on the recognition of all sentiment expressions within a given customer review and the features to which they refer. Due to the complexity of Chinese expression and the limited resources of Chinese sentiment analysis and Chinese Concept Nouns lexicon, our work performs on Chinese phone reviews faces more difficulties. A novel HL-SOT approach [6] to labeling a product’s attributes and their associated sentiments in product reviews by Hierarchical Learning (HL) process with a defined Sentiment Ontology Tree (SOT). But it needs lots of manual work to construct the SOT. A support system for Vietnamese ontology construction [7] using pattern-based mechanisms to discover Vietnamese concepts and conceptual relations from Vietnamese text documents. They used the combination of statistics-based, data mining and natural language processing methods to develop concept and conceptual relation extraction algorithms to discover knowledge from text documents. A method [8] analyze the influence of the hierarchical relationship between the product attributes and their sentiments on the overall review polarity. Moreover, they used ConceptNet to automatically create a product specific ontology with feature-specific polarities which are aggregated bottom-up. However, ConceptNet hasn’t so much information and concepts in Chinese. A novel method [9] that integrates domain sentiment knowledge into the analysis approach to deal with feature-level opinion mining by constructing a domain ontology called Fuzzy Domain Sentiment Ontology Tree (FDSOT). They utilize the prior sentiment knowledge of ontology to achieve significantly accuracy in sentiment classification. In this paper, our proposed system can semi-automatically build our ontology using Protégé for representing product features in reviews. Note that we selected features from the pre-processing process and then return to original sentences to find relationships. Finally we use the well-defined pattern to find more relationships enriching ontology and take words from online

Chinese lexicon. Not only do we decrease the risk of inaccuracy in human-define, but also we mining more information from the source reviews data.

## Methodology

Feature Ontology Based Product Review Miner (FOBPRM) system consists of three parts. First of all, all the reviews data obtained must be pre-processing. Then computing the term's special degree and similarity to construct an ontology tree and using lexicon to enrich it. Thirdly, using the method of information entropy to select high correlational feature-sentiment pairs and compute their polarity by frequency. Finally, summary with feature and sentiment will be output.

### Product feature extraction and selection

The "Product" here means "Phone" on the purpose of confirming comments domain. One of the most important thing is extracting features that can describe or define some attributes of the phone, such as "Power Consumption", "Home Screen". Then some selection mechanism must be done because Chinese Website comments are unconstrained and unorganized texts. So the task involves the following process.

- We crawled 36000 comments from Jing Dong website, and applied Part of Speech (POS) analysis on these sentences with the help of ICTCLAS (a widely used Chinese segmentation system). The original comments will be labeled by POS depending on whether nouns or adjectives.

- We used the well-worked programs to find out the frequency of nouns and then selected top hundreds of nouns artificially by frequency. We considered them as frequency candidate features.

- These selected nouns were divided into different categories according to the number of words. Single word nouns were treated as stop words. Double and triple words with frequency more than three hundred times and have been preliminary screened were remained. Others were classified into stop words. As for multi-words (four words and five words), we filtered the certain feature noun and added them into Candidate Relations Lexicon.

- Finally, normalization feature words that have same meanings regulated to the same specified key.

### The special degree and similarity computing of ontology tree automatic learning algorithm

Ontology is a formalized and clear description of shared concepts, these concepts can be domain nouns. It contains the description and constraint of concepts and relationship between concepts in a field. As a result, it's reasonable to construct the product feature domain.

In the ontology, we define two attributes for each feature node, which are specialty and similarity. Specialty refers to the feature's abstraction level compared to other features. It's like parent and child node in ontology, and parent node is more abstract than child node. Similarity means the similar degree between one feature others, such as the share part of parent and children nodes.

Thus, we use the statistical knowledge and reference computing method of term specialty and term similarity to automatically build the ontology tree.

- In the review text, the most direct performance of a feature's specialty reflects the relationship between the feature and its nearby words and their concurrency appearance. We are more interested in the features used to decorate or assemble other words, including adjectives and verbs.

For a specific feature, if the parent feature is more abstract than child feature, we represent it as  $d(t_i) \supset d(t_j)$  namely appearance times of  $t_i$  are much more than  $t_j$ . It can also be denoted as  $P(t_i|t_j) > P(t_j|t_i)$  from statistics.  $P(A|B)$  represents the probability that A occurs on the condition of B. If  $t_i$  is a nearby feature description word of  $t$ , then specialty of  $t$  can be denoted as:

$$Spec_{noun}(t) = \frac{\sum_{i=1}^n \left( \frac{P(t_i|t) - P(t|t_i)}{2|P(t_i|t) - P(t|t_i)| + 1} \right)}{n} \quad (1)$$

For adjectives, features with high specialty is too difficult to be decorated by adjectives. For example, "Screen" has size but "Iphone4 screen" can't be decorated by "bigger". Assume that  $t_i^{adj}$  is a nearby adjective of  $t$ , the specialty of  $t$  can be denoted as:

$$Spec_{adj}(t) = -\sum_{i=1}^n P(t_i^{adj}|t) \log[P(t_i^{adj}|t)] \quad (2)$$

For verbs, terms in a sentence usually appear as direct object of the verb. Terms with high specialty are more likely to be served as several specific verbs' direct object. Assuming that  $t_i^{verb}$  is  $t$ 's nearby verb, then the specialty of  $t$  can be denoted as:

$$Spec_{verb}(t) = -\sum_{i=1}^n P(t|t_i^{verb}) \log[P(t|t_i^{verb})] \quad (3)$$

➤ There are various algorithm computing term similarity at present, so we adopt mixed method of symmetric and asymmetric similarity calculation.

Symmetric similarity calculation is simply compute the two concurrently appearance frequency, formula is as follows:  $Sim_{sym}(t_i, t_j) = \frac{2 \times d(t_i, t_j)}{d(t_i) + d(t_j)}$ ,  $d(t_i, t_j)$  represents the times that  $t_i$  and  $t_j$  occur at the same time.

If we use above algorithm computing term similarity, the result of  $Sim(t_i, t_j)$  and  $Sim(t_j, t_i)$  are the same. However, the result of asymmetric algorithm is different to obtain. Follows are kernel of this algorithm:

$$Sim_{asym}(t_i, t_j) = \frac{\sum_{k=1}^n w_{ijk}}{\sum_{k=1}^n w_{ik} \times WFactor(t_j)} \quad (4)$$

a)  $w_{ijk}$  represents combined weights of  $t_i$  and  $t_j$  in review  $k$ . Among them,  $tf_{ijk}$  means frequency of term  $t_i$  and  $t_j$  appearing at the same time,  $df_{ij}$  represents numbers of reviews that  $t_i$  and  $t_j$  appearing concurrently,  $m_{t_it_j}$  represents words numbers of  $t_i$  and  $t_j$ ,  $n$  is number of reviews.

$$w_{ijk} = tf_{ijk} \times \log\left(\frac{n}{df_{ij}} \times m_{t_it_j}\right) \quad (5)$$

b)  $w_{ik}$  express term  $t_i$  weight in comment text  $k$ :

$tf_{ik}$  is the appearance frequency of  $t_i$  in  $k$ ,  $df_i$  is the number of texts which include term  $t_i$ .  $m_{t_i}$  is the number of words in term  $t_i$ .

$$w_{ik} = tf_{ik} \times \log\left(\frac{n}{df_i} \times m_{t_i}\right) \quad (6)$$

c)  $WFactor(t_j)$  is defined as:

$df_j$  is texts number including  $t_j$ ,  $WFactor(t_j)$  is a weight function which filter common vocabulary, which is equal to the inverse document frequency.

$$WFactor(t_j) = \frac{\log\left(\frac{n}{df_j}\right)}{\log n} \quad (7)$$

Finally, the final similarity is a combination of the above two algorithm.

$$Sim_{total}(t_i, t_j) = \alpha \times Sim_{sym}(t_i, t_j) + \beta \times Sim_{asym}(t_i, t_j) \quad (8)$$

$$(\alpha + \beta = 1)$$

### Pairs of feature-sentiment correlation calculation based on information entropy [10]

There are lots of features and sentiments words in the feedback comments. However the computer can't easily recognize which sentiments and features are in pair. From the statistical point of view, when the number of the comment text itself tends to be huge, users' description in corresponding will tend to concentrate on a fixed emotional words set. Then the frequency of sentiment and feature will gradually tend to reach a steady state, which performs in a variety of mixed probability.

We focus on the occurrence probability at the same time and the probability of non-occurrence frequency simultaneously when the frequency is not that high. When the number of different emotion words corresponding to one feature is large, there is a need to consider the frequency of one occurrence while another one don't.

Due to the fact that emotional words usually have many synonyms in the text, when calculating the correlation we regard the emotional word set as a whole for convenience. Following formulas use emotional words instead of nearly righteousness word set.

Definition	Expression
The occurrence times of feature $f$ , sentiment $s$ in comments	$d(f), d(s)$
The simultaneously occurrence times or non-occurrence times of $f, s$	$d(f, s), d(\bar{f}, \bar{s})$
One occurrence, the other don't	$d(f, \bar{s}), d(\bar{f}, s)$
Sentiment $s$ and feature $f$ frequency	$P(s) = \frac{d(s)}{N}, P(f) = \frac{d(f)}{N}, N$ is number of reviews
Joint frequency of $f$ and $s$	$P_r(f, s), P_r(\bar{f}, \bar{s}), P_r(\bar{f}, s), P_r(f, \bar{s})$
Positive and negative adjustment factor to prevent bad influence due to low word frequency	$\varepsilon$ ( <i>positive</i> $\in (0.1, 0.3)$ ), $\rho$ ( <i>negative</i> $\in (0.5, 0.7)$ )

$$P_r(f, s) = \log_2 \frac{d(f, s)}{(N+1) \times P(s) \times P(f)} \quad (9)$$

$$P_r(\bar{f}, \bar{s}) = \log_2 \frac{d(\bar{f}, \bar{s})}{(N+1) \times (1-P(s)) \times (1-P(f))} \quad (10)$$

$$P_r(f, \bar{s}) = \log_2 \frac{d(f, \bar{s})}{(N+1) \times (1-P(s)) \times P(f)} \quad (11)$$

$$P_r(\bar{f}, s) = \log_2 \frac{d(\bar{f}, s)}{(N+1) \times P(s) \times (1-P(f))} \quad (12)$$

Formula of Correlation calculation

$$W_{rd}(f, s) = P_r(f, s) + \varepsilon \times P_r(\bar{f}, \bar{s}) - \rho \times (P_r(f, \bar{s}) + P_r(\bar{f}, s)) \quad (13)$$

In order to make  $W_{rd}$  in the domain of  $[0, 1]$ , iterative calculate each of the candidate emotional word  $s$  for every single feature  $f$ .

$$W_{rd}'(f, s) = \alpha \times W_{rd}(f, s) + \beta \times \frac{W_{rd}(f, s) - \min\{W_{rd}(f, s_i)\}}{\max\{W_{rd}(f, s_i)\} - \min\{W_{rd}(f, s_i)\}} \quad (14)$$

$\alpha, \beta$  are iteration speed regulating factors.

To obtain the feature and sentiment pairs from the ontology is relatively complete. The actual frequency is not ideal in Chinese comments of a product, so there will be a certain default value in the result data. Since different probability account for different degrees to the feature-sentiment pair connection, we can choose the high correlation matching pair, and set the selection probability with priority so as to get a better match even if we have less parameters.

### Frequency Polarity calculation of feature-sentiment pairs

Information Gain is a widely used feature selection methods in machine learning. From the perspective of Information theory, it divide learning sample space according to the feature value, considering the size of the corresponding Information obtained. In Information theory, the size of the quantity of information is known as "entropy".

Assume that variable  $x$  has  $\{x_1, x_2, \dots, x_n\}$ , total kinds of  $n$  values, and the corresponding probability of each value is  $\{P(x_1), P(x_2), \dots, P(x_n)\}$ , then the entropy of variable  $x$  can be defined as:

$$H(x) = -\sum_{i=1}^n P(x_i) \log_2 P(x_i) \quad (15)$$

To produce a reasonable and directional summary, we need to judge on the extracted feature-sentiment pairs. We extract both the user's favorable comments and negative comments,

Then collect the probability of pair to get user's expression of emotion tendency. Assumption  $d_{pos}$  is a positive text,  $d_{neg}$  represents negative text,  $t_{fs}$  is the feature-sentiment pair, so the positive review probability of feature-sentiment is:

$$P_{pos}(t_{fs}) = \frac{P(d_{pos}, t_{fs})}{P(d_{pos})P(t_{fs})} \quad (16)$$

Similarly, for the probability of negative comments is:

$$P_{neg}(t_{fs}) = \frac{P(d_{neg}, t)}{P(d_{neg})P(t_{fs})} \quad (17)$$

We can calculate the polar expressions of feature-sentiment pair by means of information gain, among them and adjust factor to prevent the influence by less number of unilateral comments. It's influence is covered by the other side, and the value of both is determined by the number of positive and negative text ratio and the experimental results.

$$W_{pd}(t_{fs}) = A \times P_{pos}(t_{fs}) \log_2[P_{pos}(t_{fs})] - B \times P_{neg}(t_{fs}) \quad (18)$$

In order to make  $W_{pd}$  in the domain of  $[0,1]$ , iteratively calculate each of the feature-sentiment pair as follows:

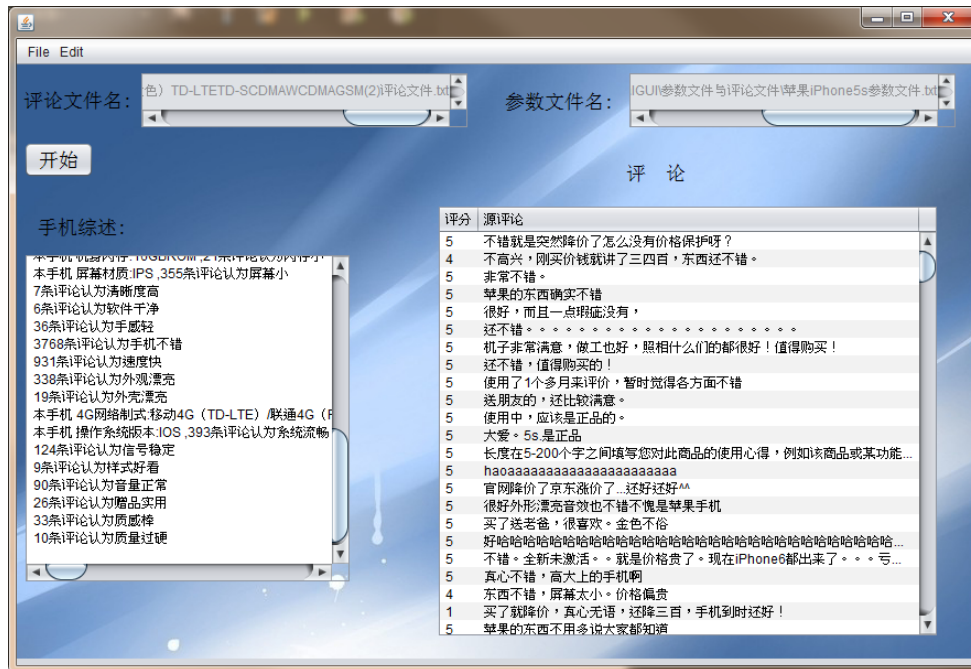
$$W_{pd}'(t_{fs}) = \alpha \times W_{pd}(t_{fs}) + \beta \times \frac{W_{pd}(t_{fs}) - \min\{W_{pd}(t_{fs})\}}{\max\{W_{pd}(t_{fs})\} - \min\{W_{pd}(t_{fs})\}} \quad (19)$$

## Test results

### A case study

FOBPRM is a review mining system which aims to extract the most representative product features and corresponding sentiments that applies the method described in this paper, finally generate a summary for people.

First of all, we use a case study to demonstrate the review summaries generated by our systems. In this case study, we use the crawler to get the original product reviews from Jd.com website, mainly 36000 reviews in phones. Then we use these data to construct our ontology and enrich it. Finally, We input Iphone 5s users' comments data into this system, and the output is Iphone's summary.



## Ontology and feature-sentiment extraction evaluation

On the basis of ontology tree, randomly for about 20000 low moisture comments, calculating the high correlation feature-sentiment pairs have an average of 36, and the accurate analysis has the mean value of 30. If remove 0 co-occurrence invalid pair for example, the "high safety", the actual accuracy will reach more than 90%. As a result, the correlation algorithm we applied has high accuracy.

Number of Reviews	Feature-sentiment on Ontology	High correlation pairs	Invalid pairs	Right pairs	Accuracy
24327	204	39	6	31	94%
20242	197	35	4	30	97%
21658	174	33	5	27	96%

### Feature-sentiment polarity evaluation

Similarly, after testing the computation of polarity, take the lowest accuracy of the results, then can get the following test values:

	Accuracy	Recall rate	FA value
Correlation algorithm	94.00%	89.23%	91.55%
Polarity calculation algorithm	80.91%	78.96%	79.92%

It's important to note that we only statistics reviews that contains at least contains an item of feature and sentiment and all comments after simply filtering to compute the recall rate, so there untreated original reviews data should be at about 20%, including statement which didn't mention any features and only have such as "good" "Not bad" generic comment statements.

Later we can improve the integrity of the ontology tree to improve the recall rate of the original incomplete or damaged emotional comments by properly completing.

### Conclusion

In this paper, we propose an ontology-based review summarization system, to automatically extract the most representative products' feature-sentiment pair and customer opinions in the reviews of the phone. The system make improvements on product feature extraction and sentiment identification by algorithm of information entropy, and feature based ontology construction by algorithm of special degree and similarity computing. The selected sentences represent the expressions and customer opinions in the product reviews on various product features. Comprehensive experiments and a case study demonstrate the effectiveness of the system.

### Acknowledgement

In this paper, the research was sponsored by the Student's Innovation Research Base of Beijing University of Posts and Telecommunications (Project No. XXX).

### References

- [1] Hu M, Liu B. Mining and summarizing customer reviews[C]//Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2004: 168-177.
- [2] Fellbaum C. WordNet[M]. Blackwell Publishing Ltd, 1998.

- [3] Cui H, Mittal V, Datar M. Comparative experiments on sentiment classification for online product reviews[C]//AAAI. 2006, 6: 1265-1270.
- [4] Dave K, Lawrence S, Pennock D M. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews[C]//Proceedings of the 12th international conference on World Wide Web. ACM, 2003: 519-528.
- [5] Agarwal A, Biadys F, Mckeown K R. Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams[C]//Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2009: 24-32.
- [6] Wei W, Gulla J A. Sentiment learning on product reviews via sentiment ontology tree[C]//Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010: 404-413.
- [7] Nguyen B A, Yang D L. A semi-automatic approach to construct Vietnamese ontology from online text[J]. The International Review of Research in Open and Distributed Learning, 2012, 13(5): 148-172
- [8] Mukherjee S, Joshi S. Sentiment aggregation using conceptnet ontology[C]//Proceedings of the 6th International Joint Conference on Natural Language Processing, IJCNLP. 2013.
- [9] Nie X, Liu L, Wang H, et al. The Opinion Mining Based on Fuzzy Domain Sentiment Ontology Tree for Product Reviews[J]. Journal of Software, 2013, 8(11): 2682-2687.
- [10] Lau R Y K, Lai C C L, Ma J, et al. Automatic domain ontology extraction for context-sensitive opinion mining[J]. ICIS 2009 Proceedings, 2009: 35-53.